Estimating Public Opinion in Social Media Content Using Aspect-Based Opinion Mining

Yen Hong Tran[1(✉)] and Quang Nhat Tran[2]

[1] People's Security Academy, Hanoi, Vietnam
yenth.hvan@gmail.com
[2] University of New South Wales at ADFA, Canberra, Australia
quang.tran@student.unsw.edu.au

Abstract. With the development of the Internet, social media has been the main platform for human to express opinions about products/services, key figures, socio-political and economic events… Besides the benefits that the platform offers, there are still various security threats relating to the fact that most extremist groups have been abusing social media to spread distorted beliefs, to incite the act of terrorism, politics, religions, to recruit, to raise funds and much more. These groups tend to include sentiment leading to illegal affairs such as terrorism, cyber-attacks, etc. when sharing their opinions and comments. Therefore, it is necessary to capture public opinions and social behaviors in social media content. This is a challenging research topic related to aspect-based opinion mining, which is the problem of determining what the exact opinions on specific aspects are rather than getting an overall positive or negative sentiment at the document level. For an entity, the main task is to detect all mentioned aspects of the entity and then produce a summary of each aspect's sentiment orientation. This paper proposes an aspect-based opinion mining model to address the problem of estimating public opinion in social media content. The model has two phases: 1 - extracting aspects based on double propagation techniques, and 2 - classifying opinions about the detected aspects with the consideration of the context of review sentences using the hybrid approach of machine learning and lexicon-based method.

Keywords: Aspect-based opinion mining · Aspect extraction
Sentiment orientation · Public opinion analysis · Natural language processing
Text mining · Social behavior

1 Introduction

With the proliferation of the Internet, massive user-generated content is posted in blogs, review sites, and especially social networks like Facebook, Twitter. The unprecedented volume as well as variety of user-generated content brings about new opportunities to understand social behavior and build socially-aware systems. This kind of data with subjective nature indicates public opinion. Public opinion influences and provides guidance for individuals, organizations, governments, and social communities during the decision-making process. While customer reviews might be useful for product sales and business, blogs and social networks can be used for political, religious, and security

J. Hu et al. (Eds.): MONAMI 2017, LNICST 235, pp. 101–115, 2018.
https://doi.org/10.1007/978-3-319-90775-8_9

issues. For example, messages in blogs that express social resentment at high intensity levels could be flagged as possible terrorist threats. Therefore, there exists an obligation to detect and categorize the opinions in social media to predict the user interest or behavior towards a specific domain, such as e-commerce, politics, security… This challenging task has foundations of natural language processing and text mining referred to opinion mining or sentiment analysis [1].

1.1 Opinion Mining

Khan et al. [2] states that an opinion represents the ideas, beliefs, and evaluations about a specific entity such as an event, a product, an organization or an individual. An opinion can be expressed in a variety of ways and generally has three main components: the source of the opinion (the opinion holder), the target of the opinion (the object about which opinion is expressed), and the opinion itself. It is simply a positive, negative or neutral view about an entity or an aspect of the entity from an opinion source. Positive, negative and neutral are called opinion orientation, sentiment orientation, semantic orientation or polarity. Opinion mining or sentiment analysis can be seen as the computational study of opinions, attitudes, and emotions toward entities and their different aspects [3]. Opinion mining has been an active research topic of knowledge discovery and data mining (KDD) in recent years due to its wide range of applications and many challenging research problems. Besides a variety of practical applications in commercial area such as summaries of customer's reviews, recommendation systems…, one of its potential application can be in political and security domain, such as internet public opinion monitoring and analyzing systems to help government intelligently understand, monitor sensitive public opinion and guide them [4]. Opinion mining can be used to examine social media networks to detect cyberbullying [5–7] or discussions concerning resentment society or planned criminals such as cyberattacks [8] with sophisticated attacker techniques and potential victims. Some recent research works have focused on applying opinion mining to detect security threats, such as terrorism [9, 10].

1.2 Aspect-Based Opinion Mining

Basically, there are three levels of opinion mining which have studied in the past decade (document level, sentence level and aspect level). Although opinion mining at document level and sentence level can be helpful in many cases, to obtain more fine-grained opinion analysis, it is necessary to delve into aspect level because positive (negative) evaluative text on an entity does not mean that the author has positive (negative) opinions on every its aspects. Aspect-based opinion mining provides opinions or sentiments about various aspects of a specific entity and entity itself. It was first called "feature-based opinion mining" in [11]. The basic task of aspect-based opinion mining is to extract aspects and summarize opinions expressed on aspects of entities. To mine opinion at aspect level, there are two core sub-tasks: 1 - extracting aspects of the entities in evaluative texts and 2 - determining sentiment polarities on aspects of entities.

The paper is organized as follows. In Sect. 2, we review and analyze some examples of previous work on aspect extraction and sentiment classification. We then describe our proposed method for aspect extraction and sentiment analysis in Sect. 3. Section 4 contains evaluation of a case study in e-commerce domain. Finally, Sect. 5 draws conclusions and examines possibilities of future work.

2 Related Work

Hu and Liu [12] first proposed an unsupervised learning method based on association rules to extract product's aspects. The main idea of this technique is that users often use the same words for a specific aspect in their comments. Therefore, the frequent item sets which are nouns and noun phrases in the evaluative text are more likely to be the product's aspects. Input of Hu and Liu's aspect extraction model is a dataset of product's reviews. This dataset is transmitted to the extraction module after the preprocessing step (split sentences, part-of-speech tagging). The result obtained is a set of frequent aspects which are evaluatively mentioned by many reviewers ("frequent" means appearing in the dataset at a frequent rate greater than a determined experimental threshold). Based on this result, the system extracts evaluative words (opinion words) and detects infrequent aspects (with small number of occurrences). Aspect extraction method based on frequent item sets that Hu and Liu proposed requires a massive volume of reviews. However, extraction process still generates much noise, such as nouns or noun phrases which are frequent in both dataset and general language.

The method of Popescu and Etzinoni [13] is based on a similar idea of Hu and Liu [12]. However, their proposed technique can eliminate frequent phrases which are most likely not to be aspect expression based on the name of entity and Pointwise Mutual Information (PMI) score between the frequent phrases and the part-whole patterns like "of xx", "xx has", "xx comes with"…, in which "xx" is a word or phrase of entity. However, PMI copes with the problem of sparsity because bigrams composed of low-frequency words might receive a higher score than those composed of high frequency words. The extraction system also costs considerable time to incorporate the Web PMI statistics to review data in its assessment.

Qiu et al. [14] proposed double propagation algorithm. The idea of this approach is based on dependency relations between opinion words and aspect expressions. The opinion-aspect relationship is determined by a dependency parser. Knowing dependency relation and one of the two components (aspect expression or opinion word), the system can detect the remaining component. The extracted opinion words and aspects are then utilized to identify new opinion words and new aspects, which are used again to extract more opinion words and aspects. This process was repeated until no more opinion words or aspect expressions can be found. This algorithm is called double propagation because information spreads between opinion words and aspects after each iteration. Besides, this approach is also considered as a semi-supervised learning method because a small number of initial seeds are used to start the process of propagation. The effectiveness of this method depends on the selection of seeds at the initial step. In [14], initial seeds are randomly selected from an available list of opinion words. Thus, in the case, if there are no opinion seeds can be found from the evaluative text, the extraction will be ineffective. In addition, the propagation based on the syntactic rules is still generated much noise if the size of dataset is large. This requires an effective method of noise removal to improve the accuracy.

In aspect-based opinion mining, after extracting aspect candidates from the evaluative dataset, the problem is to generate opinion summary for each aspect. However, users can use different words or phrases to mention one aspect, for example, “picture” and “image” are two different words but indicate the same aspect. Therefore, to create a meaningful summary, different expressions of one aspect should be grouped. There have been many methods proposed to solve this problem [15–17]. The key element of these learning algorithms is similarity score. There are two main approaches for similarity score, including: dictionary-based/lexical similarity and corpus-based/distributional similarity.

The other main task of opinion mining is sentiment orientation. Sentiment orientation is used to classify aspects, sentences or documents as positive, negative or neutral. Positive/negative polarity means that the opinion holder’s statement shows a positive/negative attitude toward the target object/aspect. Sentiment classification techniques can be divided into two categories: 1 - machine learning approach, and 2 - lexicon based approach [4]. Machine learning approach has the foundation of machine learning algorithms and linguistic features. The most frequently used algorithms for supervised sentiment classification are support vector machines (SVM), Naive Bayes classifier and Maximum entropy. Pang et al. [18] firstly adopted this approach to classify sentiment of movie reviews, however, they showed that the three machine learning methods they employed (Naive Bayes, maximum entropy classification, and support vector machines) do not perform as well on sentiment classification as on traditional topic-based categorization. The lexicon-based approach relies on a sentiment lexicon and is divided into dictionary-based approach [11] and corpus-based approach [19] which use statistical or semantic methods to find sentiment polarity. The dictionary-based approach finds opinion words, and then searches the dictionary of their synonyms and antonyms, therefore, it has a major disadvantage which is the inability to find opinion words in specific context domain. The corpus-based approach begins with a list of opinion seeds, and then finds other opinion words in a large corpus to solve the problem of context specific orientations. However, it is not a trivial task to prepare a such huge corpus.

3 Proposed Aspect-Based Opinion Mining Model

We use the term *entity* to denote the target object that has been evaluated. An entity can be represented as a tree and hierarchically decomposed based on the part-of relation. The root of the tree is the name of the entity. Each non-root node is a component or subcomponent of the entity. Each link is a part-of relation. Each node is associated with a set of attributes. An opinion can be expressed on any node and any attribute of the node [3]. To simplify, we use the term *aspects* to denote both components and attributes.

Each entity E is represented with a finite set of aspects $A = \{a_1, a_2 \ldots a_n\}$ and each aspect a_i in A can be represented by a finite set of aspect expressions AE_i. A word or a phrase $ae_{ik}\left(1 \leq k \leq |AE_i|\right)$ in AE_i will be mentioned in a review sentence s_j and opinion orientation about aspect a_i in the sentence s_j will be expressed by using opinion expressions $oe_{ijh} \in OE_{ij}$, OE_{ij} is a finite set of opinion expressions in sentence s_j for aspect $a_i \left(1 \leq h \leq \left|OE_{ij}\right|\right)$. The objective of aspect-based opinion mining is to extract and group all phrases ae_{ik} in one aspect a_i, for each review sentence s_j discover all tuples (a_i, oe_{ijh}, s_j), and finally generate an aggregated opinion summary for each aspect a_i through all review sentences s_j.

After studying some related research, we choose the extraction method based on the approach of Qiu et al. [14]. However, instead of semi-supervised learning with initial seeds of opinion words, we propose to use aspect seeds which are automatically selected from the input dataset with the orientation of a human-defined aspect sample. The human-defined aspect sample is domain-dependent and provided as the supplemental input of the system. To eliminate incorrect detected aspect candidates, the system has further steps that group aspect expressions ae_{ik} in each appropriate aspect node a_i. All tuples (a_i, oe_{ijh}, s_j) discovered from double propagation process will be assigned a sentiment orientation label using the hybrid approach of machine learning (Naïve Bayes classifier) and lexicon-based methods (Wordnet dictionary) with context consideration (dependency relations in each review sentences). An opinion summary for each aspect a_i of an entity E from input dataset will be generated as finally result.

Suppose that input dataset has been already collected and preprocessed, we propose an aspect-based opinion mining model with two phases: 1 - aspect and opinion word extraction, 2 - aspect-level sentiment classification and summary.

3.1 Aspect and Opinion Word Extraction

a. ***Generating aspect seeds***

Generating aspect seeds is performed as follows: For a specific entity domain, there is a human-defined aspect sample playing role as the input of the module. Each phrase of this sample is split into individual word. The system searches for the appearances of these words in the input review text using simple string matching. The words appear to be aspects should be nouns. With “optical zoom”, a human-defined aspect in camera domain, for instance, the system searches for word “zoom” in the review texts and obtains noun phrases containing “zoom” as the potential aspect expressions (Fig. 1).

body	
	size
	weight
	design
image	
	image type
	resolution
storage	
	storage size
	storage type
editing	
	screen size
	viewfinder type
	display
lens	
	optical zoom
	digital zoom
	zoom range
battery	
	battery life
sensor	
	sensor type
	sensor size

Fig. 1. A human-defined aspect sample in camera domain

b. *Double Propagation*

With the aspect seeds extracted previously, the system continues to expand the aspect set through the process of double propagation algorithm. Denote OA-Rel for the relationship between opinion words and aspects, OO-Rel for the relationship between opinion words and AA-Rel for the relationship between the aspects.

In double propagation algorithm [14] there are four sub-steps: (1) extract aspects using opinion words and OA-Rel relationship, (2) extract aspects using aspects and AA-Rel relationships, (3) extract opinion words using aspects and OA-Rel relationship, (4) extract opinion words using opinion words and OO-Rel relationship.

The input of the algorithm is aspect seeds A and evaluative dataset R. The processing steps in the algorithm are presented in detail in Fig. 2. The loop stops when not find any new aspects or opinion words. Here, we analyze an example to clarify the steps in the algorithm. Considering the following review:

> *"Canon G3 gives great picture. The picture is amazing. You may have to get storage to store high quality pictures and recorded movies. And the software is amazing."*

Suppose that the input of the algorithm has only one aspect as "picture". In the first iteration, executing the command line 4 will extract opinion words "great" and "amazing", then after the command line 5 executes we get "movies" as an aspect, performing the command line 11, we get aspect "software". Finally, iterative process stops because there is no more any aspects or opinion words found. Thus, through the double propagation from an initial aspect seed, two other aspects and two opinion words detected.

```
Input: Aspect seeds A {aspectSeeds},
          Evaluative dataset R
Output: Set of extracted aspects {aspectEx}
           Set of extracted opinion words {opinionEx}
Algorithm: double propagation
1. {aspectEx} = {aspectSeeds};
2. {opinionStepi}=∅;  {aspectStepi}=∅; {opinion}=∅; {aspect}=∅;
3. for each sentence s in R
4.        extract {opinionStepi} based on {aspectEx} using OA-Rel;
5.        extract {aspectStepi} based on {aspectEx} using AA-Rel;
6. endfor
7. set {opinionEx} = {opinionEx} + {opinionStepi};
8. set {aspectEx} = {aspectEx} + {aspectStepi};
9. for each sentence s in R
10.      extract {opinion} based on {opinionStepi} using OO-Rel;
11.      extract {aspect} based on {opinionStepi} using OA-Rel;
12. endfor
13. set {aspectStepi} = {aspectStepi} + {aspect};
14. set {opinionStepi} = {opinionStepi} + {opinion};
15. set {aspectEx} = {aspectEx} + {aspect};
16. set {opinionEx} = {opinionEx} + {opinion};
17. repeat 2 until (size{aspectStepi} = 0) and (size{opinionStepi} = 0);
```

Fig. 2. Double propagation algorithm

c. *Grouping aspects*

Aspect candidates obtained from the previous steps are likely to contain a lot of redundancy. In this step, aspect grouping is to reduce redundancy and based on lexical similarity in WordNet [20] and a hierarchical structure of a human-defined aspect sample which is domain-dependent. Aspect grouping has some benefits: 1 - aspects are grouped into a hierarchical structure, for example, "weight" and "size" are grouped under father node "body"; 2 - reduce redundancy, for example, "picture", "image", "image quality" are grouped in one aspect node "image".

The task of grouping aspects is equivalent to mapping each phrase in the set of extracted aspect candidates (AC) to a node in the human-defined aspect structure AT. The mapping process is performed based on phrase similarity metrics which are calculated from word similarity metrics.

- *Word similarity metrics*

Denote c_i and t_j are the corresponding phrases of the AC and AT, respectively

– Simple string matching

$$str\_match(c_i, t_j) = \begin{cases} 1 \ \ if\ c_i\ match\ t_j \\ 0\ \ if\ c_i\ do\ not\ match\ t_j \end{cases}$$

– Use information from WordNet and the type of word (part of speech).

In WordNet, each word is grouped into one or more synonymous sets called *synset* based on part-of-speech tags and semantics of the words. Each synset is a node in the taxonomy of the WordNet. If a word has more than one meaning, it will appear in many synsets at many positions in the taxonomy. Function $syns(w)$ returns a set of all synsets that w belongs.

$$syn\_score(c_i, t_j) = \begin{cases} 1 & if\ syns(c_i) \cap syns(t_j) \neq \emptyset \\ 0 & if\ syns(c_i) \cap syns(t_j) = \emptyset \end{cases}$$

– Use some similarity measure *sm* introduced in [21]

Measuring semantic similarity between two synsets in the taxonomy WordNet has two approaches: the first is based on the distance between two nodes of the taxonomy corresponding to the two synsets, the second is relied on shared information of the two synsets which is the content of the nearest common parent node of them. Here, we use the second approach.

$$sym\_score_{sm}(c_i, t_j) = \frac{sm(c_i, t_j)}{max(sm)}$$

$sm(c_i, t_j)$ can be calculated from one of the following expressions:

$$sm(c_i, t_j) = Res(c_i, t_j) = IC(LCS(c_i, t_j))$$

$$IC(c_i) = -log\,Pr(c_i)$$

$$sm(c_i, t_j) = Lin(c_i, t_j) = \frac{2 \times Res(c_i, t_j)}{IC(c_i) + IC(t_j)}$$

$$sm(c_i, t_j) = Jcn(c_i, t_j) = \frac{1}{IC(c_i) + IC(t_j) - 2 \times Res(c_i, t_j)}$$

$IC(w)$ is the information content (*IC - Information Content*) of node w in WordNet.

$LCS(w_1, w_2)$ is the nearest common node (LCS - Least Common Subsume) of w_1 and w_2 in WordNet.

$Pr(w)$ is the probability of word w appear in the dictionary WordNet.

$Res(w_1, w_2)$, $Lin(w_1, w_2)$, $JCN(w_1, w_2)$ are the types of semantic similarity between w_1 and w_2.

- *Phrase similarity metrics*

Denote ac_i and at_j are phrases in AC and AT respectively. c and t are the corresponding words in ac_i and at_j; *wm* stands for similarity measure between words mentioned above.

– Function *max* returns the largest similarity measure between ac_i and at_j

$$ac_i = \{c_1, \ldots, c_n\}$$

$$at_j = \{t_1, \ldots, t_m\}$$

$$max(ac_i, at_j) = max_{i,j}\{wm(c_i, t_j)\}$$

– Function *avg* returns the average similarity measure of ac_i and at_j

$$ac_i = \{c_1, \ldots, c_n\}$$

$$at_j = \{t_1, \ldots, t_m\}$$

$$avg(ac_i, at_j) = \frac{\frac{\sum_{i=1}^{n} max_j\{wm(c_i, t_j)\}}{n} + \frac{\sum_{j=1}^{m} max_i\{wm(c_i, t_j)\}}{m}}{2}$$

ac is mapped to *at* if *ac* and at has the highest similarity measure and this number is greater than a certain threshold θ. With *str_match* and *syn_score*, threshold $\theta = 0$. With *sim_score*, this threshold is set empirically.

d. ***Aspect-level sentiment orientation and summary***

Given a set of sentiment orientation (SO) labels {positive, negative, neutral} and a set of tuples (*a, o, s*), where *o* is a potential opinion word associated with aspect *a* in sentence *s*, the task is to assign an SO label to each tuple (*a, o, s*). For example, the tuple (*image, poor, I am not happy with this poor image*) would be assigned a negative.

Find an SO label for each potential opinion word *o*

Assume that semantic orientation of word *o* is the class which maximizes the probability *c* conditional on *o*, $c \in C$ and $C = \{positive, negative, neutral\}$. Every word *o* can be represented as the set of its synonyms retrieved from WordNet.

$$\begin{aligned} SO(w) &= argmax_{c \in C} P(c|o) \\ &= argmax_{c \in C} P(o|c)P(c) \\ &= argmax_{c \in C} P(syn_1, syn_2, \ldots, syn_n|c)P(c) \\ &= argmax_{c \in C} \frac{\sum_{i=1}^{n} count(syn_i, c)}{|synset_w|} P(c) \end{aligned}$$

$syn_1, syn_2, \ldots, syn_n$ are synonyms of *o* and *o* is also considered as a synonym of itself.

For a synonym syn_i, $count(syn_i, c)$ is 1 if the synonym syn_i appears with polarity *c* in the dictionary of opinion lexicon [22], otherwise it is 0. Words that cannot be found in the opinion lexicon are assumed to have neutral polarity.

Find an SO Label for Tuple (*a, o, s*) Given the *o*'s SO Label
First assign each tuple *(a, o, s)* an initial SO label which is *o*'s SO label. Then the system updates the default SO label whenever necessary based on syntactic relationships between opinion words and, respectively, between aspects. For example, (*memory, small, I hate the small memory because it shortly runs out of space*.) is a tuple detected. At the initial assignment, SO("small") = "neural". However, in the context of sentence "I hate the small memory because it shortly runs out of space.", "hate" and "small" satisfy modified rule and therefore it is expected that two these words have similar SO labels. Because "hate" is strongly negative, "small" in the context (memory, small, I hate the small memory because it shortly runs out of space.) acquires a negative SO label. To correctly update SO labels, the presence of negation modifiers is taken into consideration. For example, in the sentence "I don't like larger size because it is not convenient to handle", the positive SO label of "like" is replaced with the negative labeled and then "large" in the context of the tuple (large, size, "I don't like larger size because it is not convenient to handle") is inferred to have a negative SO label for aspect "size".

The final aspect-level sentiment of an aspect a_i in sentence s_j is determined by a simple aggregation function which sums up the semantic orientation of all opinion words oe_{ijh} from all previously detected tuples (a_i, oe_{ijh}, s_j). It is intuitive that an opinion phrase associated with an aspect will occur in its vicinity. Every semantic orientation is weighted relative to its distance to the aspect. The distance of the current opinion word and the aspect is the number of words lying in between. The idea behind this function is that opinion words which are closer to the aspect are most likely to be related to it. $score(a_i, s_j) > 0$ means that a sentiment about the aspect a_i in sentence s_j is positive, $score(a_i, s_j) < 0$ means that a sentiment about the aspect a_i in sentence s_j is negative, $score(a_i, s_j) = 0$ means that a sentiment about the aspect a_i in sentence s_j is neural [23].

$$score(a_i, s_j) = \sum_{oo_{ijh} \in s_j} \frac{SO(a_i, oe_{ijh}, s_j)}{dist(oe_{ijh}, a_i)}$$

After all the previous steps, we are simply straightforward to generate the final aspect-level review summary. For each discovered aspect of the considered entity, each sentence in the input dataset which mentions this aspect is put into positive/negative/neural classes depending on the value of $score(a_i, s_j)$. A counter is computed to show how many review sentences give positive/negative/neural opinions about this aspect.

4 Case Study

This case study examines the performance of the proposed method for the problem of estimating sentiment of online camera reviews. This set of evaluative texts is collected from the site http://epinions.com, including 347 review posts for 8 types of cameras. Reviews for the same camera are stored in the same folder (Table 1).

Table 1. Experimental dataset of eight camera types

Type	Camera name	#Post	#Sentences
C1	Canon EOS 400D	65	953
C2	Canon Power Shot A510	44	714
C3	Canon Power Shot G3	45	593
C4	Canon Power Shot S100	50	286
C5	Nikon Coolpix 4300	34	358
C6	Nikon Coolpix L6	75	1591
C7	Panasonic Lumix DMC-FX7	20	684
C8	Sony Cyber-shot DSC-H1	14	307

After processing review texts (sentences splitting, tokenizing, part of speech tagging, dependency parsing) using Html Agility Pack [24] and Stanford CoreNLP [25] we obtain linguistic information of each of 5486 sentences in dataset R. The input of system includes a human-defined aspect sample in camera domain, and the dataset R. After the phase of aspect extraction, we get a list of extracted aspects which are mapped (grouped) into appropriate aspects in the human-defined aspect sample using $sym\_score, avg, Jcn, \theta = 0.5$; and a set of tuples (a, o, s) where o is a potential opinion word associated with aspect a in sentence s (Figs. 3 and 4).

$$Precision = \frac{\#Correct\_Extracted\_Aspects}{\#Extracted\_Aspects}$$

$$Recall = \frac{\#Correct\_Extracted\_Aspects}{\#Total\_Correct\_Aspects}$$

$$F1 = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

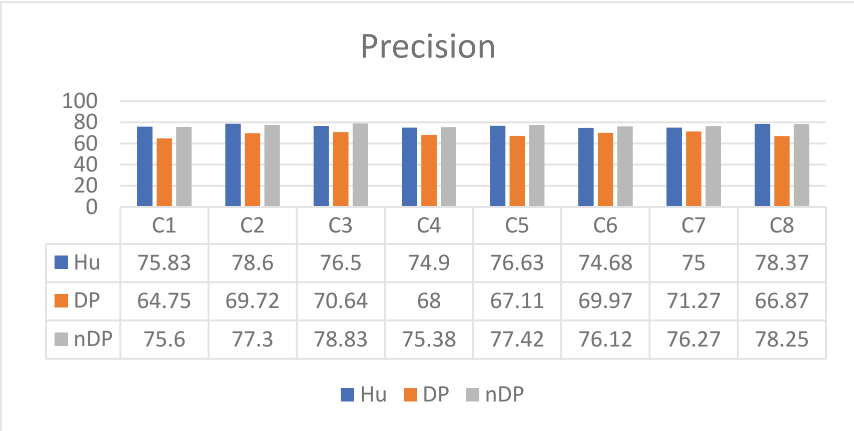

Fig. 3. Precision

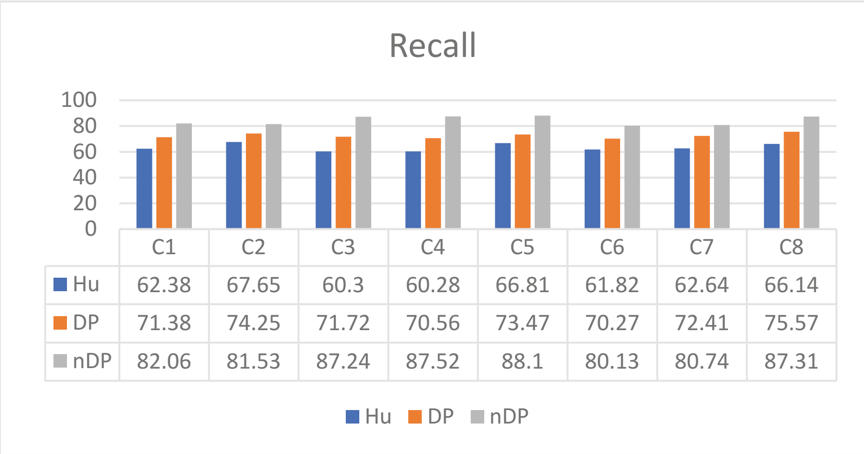

Fig. 4. Recall

See Fig. 5.

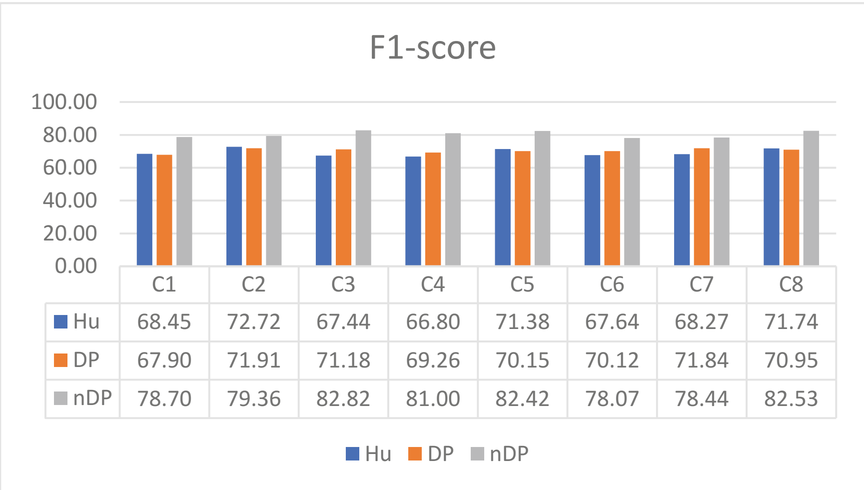

Fig. 5. F1

The charts above show the experimental results of our proposed extraction methods (nDP) compared to methods proposed by Qiu et al. [14] (DP) and methods based on the association rules of Hu and Liu [12] (Hu) in terms of precision, recall and F1 score. As can be seen from the three above charts, the precision of the proposed method (average 76.8%) is equivalent to that of Hu (average 76.3%), and both are higher than that of DP (average 68.5%). However, the recall of Hu (average 63.5%) is lower than both of DP (average 72.4%) and nDP (average 84.3%). In terms of F1 score, Hu and DP are likely to be the same results (average 69.3% and 70.4% respectively) and less effective than nDP (average 80.4%). Our result analysis indicates that Hu's method is relatively effective in extracting frequent aspects with relatively high precision, but the disadvantage is that it just successfully extracts a small number of aspects which are frequent aspects (frequent items) in total number of correct aspects which includes infrequent aspects in the dataset. The higher recall figures of DP and nDP show that these methods extract infrequent aspects better than Hu's method. Overall, the precision, recall and F1 score of the proposed nDP method are mostly higher than those of two others, indicating the effectiveness of the nDP compared to DP and Hu algorithms.

Finally, the system generates the aspect-level review summary based on an input set of tuples (*a, o, s*) found in previous aspect extraction step and the proposed phase of aspect-level sentiment orientation and summary. For each discovered aspect of camera entity, each sentence in the input dataset which mentions this aspect is put into positive/negative/neural classes depending on the value of $score\left(a_i, s_j\right)$ (Fig. 6).

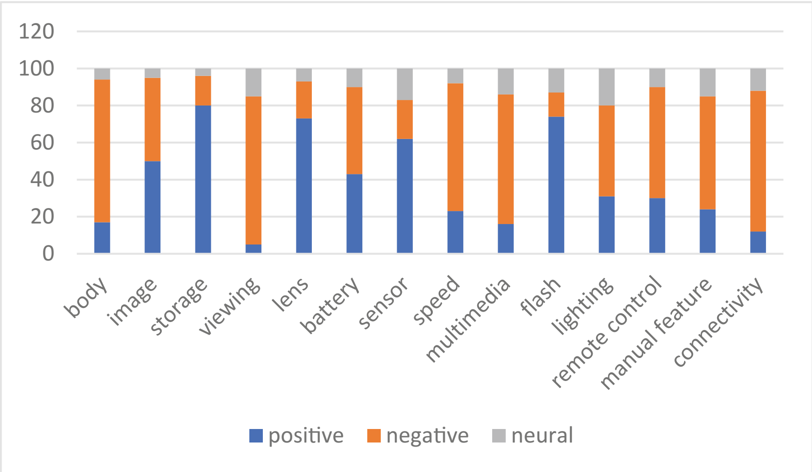

Fig. 6. Aspect-level opinion summary for experimental camera reviews

5 Conclusions

In this paper, we proposed some techniques for aspect extraction and sentiment analysis in aspect-based opinion mining problem, with the focus on: 1 - extracting both potential aspects and opinion words based on double propagation with some improvements to enhance the effectiveness of the model, 2 - classifying opinion about detected aspects in the context of review sentence using the hybrid approach of machine learning (Naïve Bayes classifier) and lexicon-based method (Wordnet) with the consideration of the sentence's context (dependency relations). Experimental results on the camera domain indicate that the proposed techniques are promising in performing the tasks of aspect-based opinion mining problem.

For future work, we plan to further improve and refine our techniques, and to address the challenging problems of determining the strength of opinions, and investigating opinions expressed with adverbs, verbs and nouns. We will also carry out more research and experiments in political and security domain.

References

1. Liu, B.: Sentiment Analysis and Opinion Mining. Morgan & Claypool Publishers, San Rafael ©2012
2. Khan, K., Baharudin, B., Khan, A.: Mining opinion components from unstructured reviews. **26**(3), 258–275 (2014)
3. Zhang, L., Liu, B.: Aspect and entity extraction for opinion mining. In: Chu, W. (ed.) Data Mining and Knowledge Discovery for Big Data. SBD 2004, vol. 1, pp. 1–40. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-40837-3_1
4. Maynard, D., Funk, A.: Automatic detection of political opinions in tweets. In: García-Castro, R., Fensel, D., Antoniou, G. (eds.) ESWC 2011. LNCS, vol. 7117, pp. 88–99. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-25953-1_8
5. Potha, N.: A biology-inspired, data mining framework for extracting patterns in sexual cyberbullying data. Knowl.-Based Syst. **96**, 134–155 (2016)
6. Zhao, R., Zhou, A., Mao, K.: Automatic detection of cyberbullying on social networks based on bullying features. In: Proceedings of the 17th International Conference on Distributed Computing and Networking, ICDCN 2016, Singapore (2016)
7. Sui, J.: Doctor of Philosophy: Understanding and fighting bullying with machine learning. University of Wisconsin-Madison (2015)
8. Lippmann, R.P., et al.: Toward finding malicious cyber discussions in social media. Presented at the The AAAI-17 Workshop on Artificial Intelligence for Cyber Security (2017)
9. Azizan, S.A., Aziz, I.A.: Terrorism detection based on sentiment analysis using machine learning. J. Eng. Appl. Sci. **12**, 691–698 (2017)
10. Wen, S., Haghighi, M.S., Chen, C., Xiang, Y., Zhou, W.L., Jia, W.J.: A sword with two edges: propagation studies on both positive and negative information in online social networks. IEEE Trans. Comput. **64**, 640–653 (2015)
11. Hu, M., Liu, B.: Mining and summarizing customer reviews. In: Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining
12. Hu, M., Liu, B.: Mining opinion features in customer reviews. Presented at the AAAI 2004 Proceedings of the 19th National Conference on Artifical Intelligence, pp. 755–760 (2004)
13. Popescu, A.-M., Etzioni, O.: Extracting product features and opinions from reviews. In: Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing, HLT 2005, pp. 339–346 (2005)
14. Qiu, G., Liu, B., Bu, J., Chen, C.: Expanding domain sentiment lexicon through double propagation. In: Proceedings of the 21st International Joint Conference on Artificial Intelligence, IJCAI 2009, pp. 1199–1204 (2009)
15. Zhai, Z., Liu, B., Xu, H., Jia, P.: Grouping product features using semi-supervised learning with soft-constraints. In: Proceedings of the 23rd International Conference on Computational Linguistics (2010)
16. Raju, S., Shishtla, P., Varma, V.: Graph clustering approach to product attribute extraction. Presented at the 4th Indian International Conference on Artificial Intelligence (2009)

17. Zhai, Z., Liu, B., Xu, H., Jia, P.: Clustering product features for opinion mining. In: Proceedings of the Fourth ACM International Conference on Web Search and Data Mining (2011)
18. Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up?: sentiment classification using machine learning techniques. In: Proceedings of the ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10, pp. 79–86 (2002)
19. Hatzivassiloglou, V., McKeown, K.: Predicting the semantic orientation of adjectives. In: Proceedings of Annual Meeting of the Association for Computational Linguistics (1997)
20. Wordnet. https://wordnet.princeton.edu/
21. Budanitsky, A., Hirst, G.: Semantic distance in wordnet: an experimental, application-oriented evaluation of five measures. Presented at the Workshop on WordNet and Other Lexical Resources (2001)
22. Hu, Liu: Opinion Lexicon: A list of English positive and negative opinion words or sentiment words. https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html#lexicon
23. Ringsquandl, M., Petković, D.: Analyzing political sentiment on Twitter. In: 2013 AAAI Spring Symposium, Stanford University (2013)
24. Html Agility Pack. http://html-agility-pack.net
25. Stanford CoreNLP. https://stanfordnlp.github.io/CoreNLP/